[19]中华人民共和国国家知识产权局

[51]Int. Cl[7]
H04L 12/42

[12]发明专利申请公开说明书

[21]申请号 00133415.8

[43]公开日 2001年4月4日

[11]公开号 CN 1290092A

[22]申请日 2000.11.3 [21]申请号 00133415.8

[71]申请人 国家数字交换系统工程技术研究中心
地址 450002 河南省郑州市俭学街7号
[72]发明人 兰巨龙 李 鸥 汪斌强

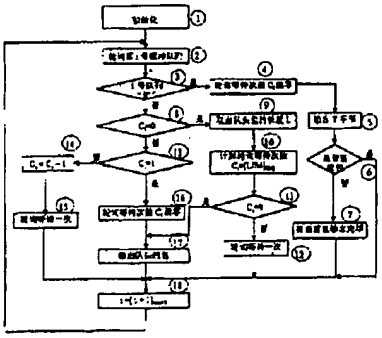
[74]专利代理机构 北京集佳商标专利事务所
代理人 王学强

权利要求书2页 说明书8页 附图页数3页

[54]发明名称 基于队列状态的累计补偿型循环轮询不定长包调度方法

[57]摘要

本发明公开了一种基于队列状态的累计补偿型循环轮询不定长包调度方法，该方法采用了轮询等待次数计数器和设置了各个输入队列的“忙”“闲”状态，在非拥塞情况下，每一队列均不能长期占用输出链路，因而某些输入队列业务量大时，也不会过分影响小业务量队列的调度，在拥塞状态，可能各个输入队列均处于“忙”状态，轮询到每个队列均调度输出有限个字节的数据，因而可快速缓解缓冲队列拥塞，可有效控制合法用户的包丢失概率。



ISSN1008-4274

知识产权出版社出版

权　利　要　求　书

1、一种基于队列状态的累计补偿型循环轮询不定长包调度方法，其特征在于，该方法包括以下步骤：

（1）初始设置循环轮询的周期 N 和初始轮询的队列号 i=0，将各个队列的轮询等待次数计数器置零，其中 N 为队列数；

（2）轮询第 i 号队列；

（3）判断第 i 号队列是否处于“忙”状态，若是，转步骤（4）继续操作，否则转步骤（8）继续操作；

（4）对轮询等待次数 C 清零；

（5）从第 i 号队列中读出 T 字节数据；

（6）判断读出的 T 字节数据是否正好为整数个包，如是，转步骤（18）继续操作，否则转步骤（7）继续操作；

（7）将当前包输出完毕，然后转步骤（18）继续操作；

（8）判断轮询等待次数计数器 C_i 是否为 0，如是，转步骤（9）继续操作，否则转步骤（13）继续操作；

（9）从第 i 队列中取出最前面一个包的包头，从中读出包长 L；

（10）计算轮询等待次数 $C_i=[L/P]_{取整}$，其中，P 为包长分界长度，包长大于 P 的分组为“长包”，包长小于等于 L 的分组为“短包”；

（11）判断 C_i 是否等于 0，如等于 0，转步骤（17）继续操作，若 C_i 不为 0，转步骤（12）继续操作；

（12）轮询等待一次，然后转步骤（18）继续操作；

（13）判断轮询等待次数计数器 C_i 是否为 1，如不为 1，转步骤（14）继续操作，否则转步骤（16）继续操作；

（14）计算 $C_i=C_i-1$；

（15）轮询等待一次，然后转步骤（18）继续操作；

（16）对轮询等待次数计数器 C_i 清零；

（17）输出第 i 号队列队头的包；

（18）计算 $i=[i+1]_{mod\ T}$，返回上述步骤（2），准备轮询下一队列。

2、根据权利要求 1 所述的基于队列状态的累计补偿型循环轮询不定长包调度方法，其特征在于，所述包长分界长度 P 根据系统环境决定。

3、根据权利要求 1 所述的基于队列状态的累计补偿型循环轮询不定长包调度方法，其特征在于，将各个队列的状态分为“忙”“闲”两种状态，在某一队列忙时，从该队列中至少读出 T 个字节，其 T 的值根据系统环境决定。

说　　明　　书

基于队列状态的
累计补偿型循环轮询不定长包调度方法

本发明涉及电信领域，特别涉及到在各种路由器、包交换系统和综合业务接入设备中不定长包的多输入、单输出排队系统。

随着通信技术的迅猛发展，各种包交换技术，特别是 IP 技术日趋成熟并实用化，良好的调度算法能够节约传输带宽，降低设备的复杂度，更重要的是能够保证网络的安全性，以防恶意破坏者的攻击。

现有的调度算法中，主要有1989年第12期数字通信特别组会议录第1到12页的题为 “公平排队策略的分析与仿真”（A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queuing algorithm. In Proc. ACM SIGCOMM'89, September 1989, pp: 1-12）公开的先到先服务策略(FCFS, First-Come-First-Serve)、公平排队策略（FQ, Fair Queuing Algorithm）和IEEE通信汇刊1987年第4期第435到438页的题为“存储器有限的包交换技术研究”（John Nagle. On packet switches with infinite storage. IEEE Trans. on Comm., COM-35(4), April 1987, pp. 435-438.）公开的逐包调度方法（Packet-by-Packet Round Robin），先到先服务策略完全不能保证各个输入流的公平性，造成网络易于受到攻击。而公平排队策略虽然能够保证各个输入流的公平性，但却非常复杂，需要*O*

（log(n)）次复杂运算才能调度一个包，其中n为经过调度系统的流数，特别不适于高速交换网络。逐包调度算法虽然运算不很复杂，也具有一定的抗恶意攻击能力，但对不定长包交换系统，对长包和短包不是同等对待，造成不公平，在极端情况下，两个同样服务等级的流，一个流所占的输出带宽可能是另一流的Max/Min倍，上述Max、Min分别为用字节表示的最大和最小包长。

针对上述现有调度方法不能兼顾公平性和复杂度的缺陷，本发明的目的是给出一种不定长包调度算法，该方法能在在低复杂度条件下，实现不定长包的公平调度。

为达到上述目的，本发明采用的技术方案是：一种基于队列状态的累计补偿型循环轮询不定长包调度方法，该方法包括以下步骤：

（1）初始设置循环轮询的周期 N 和初始轮询的队列号 i=0，将各个队列的轮询等待次数计数器置零，其中 N 为队列数；

（2）轮询第 i 号队列；

（3）判断第 i 号队列是否处于“忙”状态，若是，转步骤（4）继续操作，否则转步骤（8）继续操作；

（4）对轮询等待次数 C 清零；

（5）从第 i 号队列中读出 T 字节数据；

（6）判断读出的 T 字节数据是否正好为整数个包，如是，转步骤（18）继续操作，否则转步骤（7）继续操作；

（7）将当前包输出完毕，然后转步骤（18）继续操作；

（8）判断轮询等待次数计数器 C_i 是否为 0，如是，转步骤（9）

继续操作，否则转步骤（13）继续操作；

（9）从第 i 队列中取出最前面一个包的包头，从中读出包长 L；

（10）计算轮询等待次数 $C_i=[L/P]_{取整}$，P 为包长分界长度，包长大于 P 的分组为“长包”，包长小于等于 L 的分组为“短包”；

（11）判断 C_i 是否等于 0，如等于 0，转步骤（17）继续操作，若 C_i 不为 0，转步骤（12）继续操作；

（12）轮询等待一次，然后转步骤（18）继续操作；

（13）判断轮询等待次数计数器 C_i 是否为 1，如不为 1，转步骤（14）继续操作，否则转步骤（16）继续操作；

（14）计算 $C_i=C_i-1$；

（15）轮询等待一次，然后转步骤（18）继续操作；

（16）对轮询等待次数计数器 C_i 清零；

（17）输出第 i 号队列的队头的包；

（18）计算 $i=[i+1]_{mod\ T}$，返回上述步骤（2），准备轮询下一队列。

上述 T、P 的值根据系统环境决定。

从上述本发明采用的方法可以看出，由于本发明采用了轮询等待次数计数器和设置了各个输入队列的“忙”“闲”状态，可保证各个输入队列的公平性，在非拥塞情况下，由于每一队列均不能长期占用输出链路，因而某些输入队列业务量大时，也不会过分影响小业务量队列的调度，因为每一队列每次轮询最多输出（T+Max－1）字节的数据；在拥塞状态，可能各个输入队列均处于“忙”状态，

轮询到每个队列均调度输出 T 到（T+Max－1）字节的数据，若有恶意攻击者，只能大量丢弃自己的包；

由于设置了各个输入队列的“忙”“闲”状态，对某些输入队列的短时间突发包，由于每轮询一次可输出（T+Max－1）字节的数据，因而可快速缓解缓冲队列的拥塞。

由于限制了一个队列每次调度输出的字节数，可有效控制合法用户的包丢失概率。

由上看出，本发明在队列调度中兼顾公平性和复杂度，取得了公平性和易实现性的良好折衷，其特点是在网络不拥塞时，保证低丢包率，保证业务量大的输入队列不会影响低业务量队列的丢包率和时延。在网络拥塞时，确保 N 个缓冲队列公平地分享输出链路的容量，抗恶意攻击，因此本发明完全能够在低复杂度条件下，实现不定长包的公平调度

下面结合附图和实施例对本发明作进一步的说明。

图 1 是多输入单输出的排队模型图；

图 2 是本发明采用方法的流程图；

图 3 是本发明采用方法的原理框图；

图 4 是本发明在路由器中的应用示意图。

目前，以IP（Internet Protocol）包为代表的不定长分组业务在所有数据业务中占据越来越重要的地位，在多路接入、路由器等系统中，经常遇到多个输入、单个输出的排队模型，参考图1，由于包长是变化的，许多成熟的公平排队策略在这种情况下不能保证公

平性。为了保证各个输入端口在同一输出端口处获得公平分配带宽，同时考虑缓冲队列的情况，在各种参数的控制下，设计轮询调度控制器。

包调度算法首先提取算法必要的参数，在这些参数的控制下，对 N 个等待输出的缓冲队列中的不定长包循环输出，根据资源公平共享和尽力控制包丢失概率的宗旨，特将输出调度控制的原则确定为：确保 N 个缓冲队列公平地分享输出链路的容量；尽力缓解各缓冲队列的充盈程度；保证实时业务的低时延特性。在此条件限制下，以包丢失率最小为目标函数求得最佳控制模型。

该方法的基本原理是：首先，轮询调度控制器控制轮询机轮流访问各个输入队列，当访问到某一队列时，根据以下的调度原则从队列中调度包交给输出端。将缓冲队列的充盈程度分为“忙”和“闲”两种状态，例如，缓冲队列缓存器的剩余容量不够容纳两个长度为 $L_{长}$个字节的长包的状态定义为“忙”状态，否则为“闲”状态。将等待输出的不定长分组分为“长包”和“短包”两种，例如包长大于 P 的分组定义为“长包”，包长小于等于 P 的分组定义为“短包”，对以太网中的 IP 包，可定义 $L_{长}$=1500，P=64。所有队列的输出调度控制均基于循环轮询调度算法。当被轮询的缓冲队列处于“忙”状态时，无论其它条件如何均从该队列中至少读出 T 个字节，当读出 T 个字节时，若正好读出整数包，则轮询下一队列，否则将当前包输出完毕后再轮询下一队列。当被轮询的缓冲队列处于“闲”状态时，如果该队列最前面的分组为短包则输出该分组，否则计算出该

长包的长度相对于 P 字节的倍数 M，将 M-1 作为该长包最终输出的轮询等待次数 C，每次该队列被轮询到时 C 值减 1，直至轮询等待次数为零时才输出该长包。

上述 T、P 的值根据系统环境决定。

本发明的具体实现方法如下：

在步骤 1，初始化设置设置循环轮询的周期 N（队列数）和初始轮询的队列号 i=0，将各个队列的轮询等待次数计数器置零；在步骤 2 开始轮询，设现在轮询到第 i 号队列；在步骤 3，判断第 i 号队列是否处于“忙”状态，若是，表明该队列已经快溢出了，由步骤 4 对轮询等待次数 C 清零，在步骤 5 中从该队列中至少读出 T 个字节，当读出 T 个字节时，在步骤 6 判断是否正好读出整数包，如是，则转步骤 18 准备轮询下一队列，否则在步骤 7 将当前包输出完毕后再转步骤 18 准备轮询下一队列。

若步骤 3 判断得出第 i 号队列不处于“忙”状态，由步骤 8 判断轮询等待次数计数器 C_i 是否为“0”，若是，表明前一轮轮询时第 i 个队列没有被拖欠带宽，由步骤 9 从第 i 队列中取出最前面一个包的包头，读出包长 L，由步骤 10 计算轮询等待次数 $C_i=[L/P]_{取整}$，在步骤 11 判断 C_i 是否为零，如为零，说明当前包为短包，在步骤 17 将它调度输出，然后执行步骤 18 准备轮询下一队列。若 C_i 不为 0，说明当前包为长包，等待一轮，调度机拖欠第 i 路约 64 字节的带宽，接着执行步骤 18 准备轮询下一队列。

若由步骤 8 判断轮询等待次数计数器 C_i 不为零，表明前一轮轮

询时调度机拖欠第个 i 队列的带宽。再由步骤 13 判断轮询等待次数计数器 C_i 是否为 1，如为 1，表明前一轮轮询时调度机拖欠第个 i 队列的带宽足以将当前队列中的长包调度出去，由步骤 16 对轮询等待次数计数器 C_i 清零，然后在步骤 17 输出该队列的对头长包，最后执行步骤 18 准备轮询下一队列。

若由步骤 13 判断轮询等待次数计数器 C_i 不为 1， 表明前一轮轮询时调度机拖欠第个 i 队列的带宽不足以将当前队列中的长包调度出去，在步骤 14 再拖欠一轮，在步骤 16 轮询等待一次，再执行步骤 18 准备轮询下一队列

如此反复轮询下去，保证在无队列“忙”时，近似公平调度各个队列的包，若有恶意攻击者，则由于每次调度某一队列的字节数不超过（T+Max－1）字节，不会让其占据过多的带宽，不会影响其他队列的包传输。

上述 T、P 的值根据系统环境决定。

本发明具体实施的相应的电路实现原理框图如图 3 所示。

图中的 N 个缓存队列为输入队列，循环轮询控制电路控制轮询顺序和总线选择，确定调度输出哪一队列，是否输出该队列的包，输出一个包还是至少输出 T 个字节。包长锁存为轮询机提供各个队列处于对头等待输出包的包长。队列状态指示该队列是否处于“忙”状态，轮询等待次数锁存记录 C_i 的值。包读出控制电路与循环轮询控制电路、总线选择器共同决定当前输出哪个队列，输出多少包。图中的数据锁存为轮询机计算提供等待时间。

为了清楚地说明本发明的原理及应用，现通过在路由器中实现本发明对本发明作进一步叙述。参考图 4。图中的轮询调度模块即由本发明的算法实现。图 4 显示了多个端口的路由器框图，用户接口模块是路由器的输入输出口；转发处理模块对 IP 包继续转发，由高速交换模块按需要输出的端口交换到相应输出口的轮询调度模块，将各个输入端口传递来的包统一调度，在本端口输出；主控/管理模块通过以太网 HUB 与交换机对所有模块进行配置、管理。

在图 4 轮询调度模块的实现中，假设包长分界长度 P=64，队列数为 100，T=1500。图中的 100 个缓存队列缓存 100 个输入端口送来的包，循环轮询控制电路轮询到某一队列时，首先判断第该队列是否处于“忙”状态，若是，表明该队列已经快溢出了，从该队列中至少读出 1500 个字节，即当读出 1500 个字节时，若正好读出整数包，则轮询下一队列，否则将当前包输出完毕后再轮询下一队列。若该队列状态为“闲”，由轮询等待次数锁存器判断轮询等待次数是否为零，若是，则表明前一轮轮询时该队列没有被拖欠带宽。从该包长锁存器中读出包长 L，计算轮询等待次数 $C_i=[L/64]_{取整}$，若轮询等待次数锁存器为 0，说明当前包为短包，将它调度输出，再轮询下一队列。若轮询等待次数锁存器为不为 0，说明当前包为长包，需要等到第 C_i 轮才能输出该包。

说　明　书　附　图

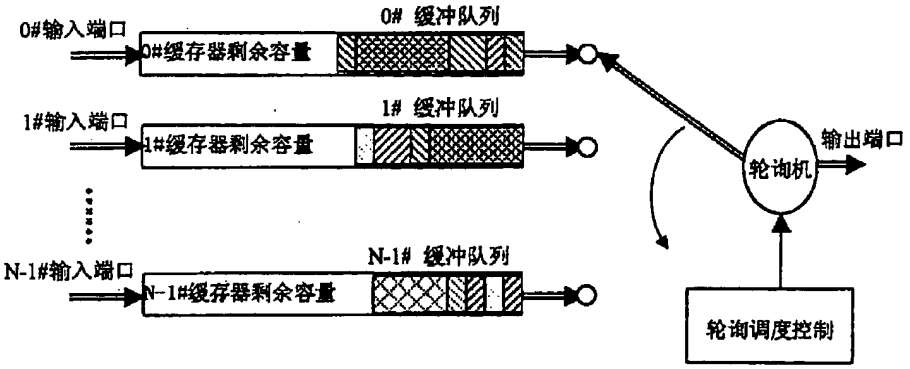

0#输入端口
0# 缓冲队列
0#缓存器剩余容量
1#输入端口
1# 缓冲队列
1#缓存器剩余容量
N-1#输入端口
N-1# 缓冲队列
N-1#缓存器剩余容量
轮询机
输出端口
轮询调度控制

图 1

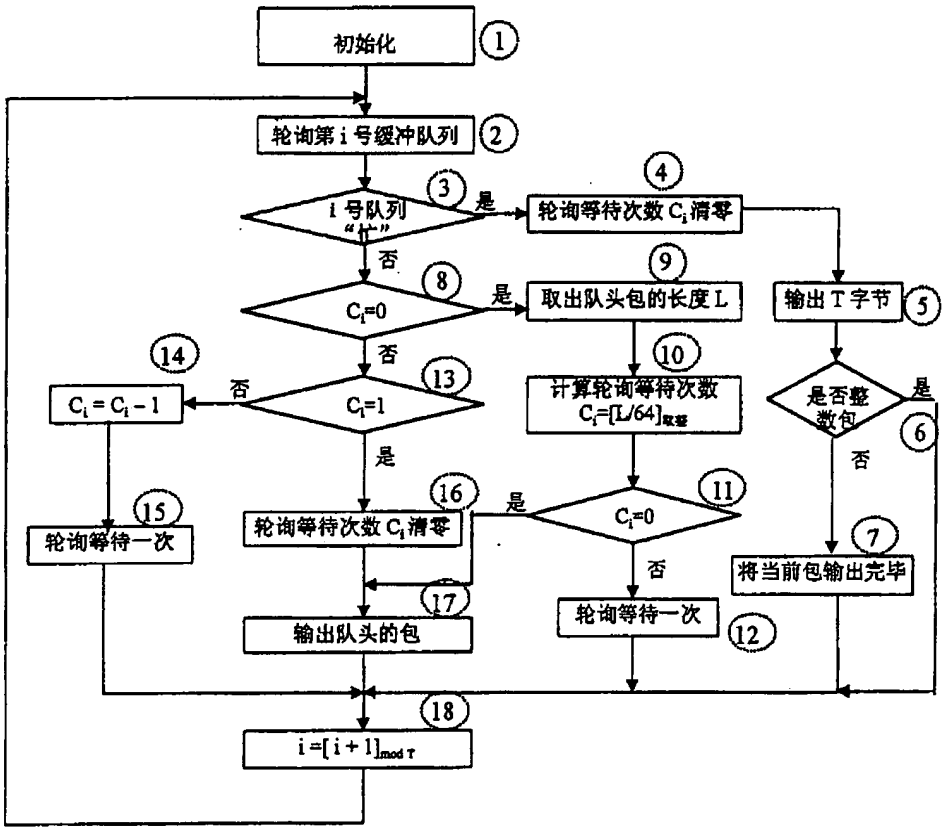

初始化
1
轮询第 i 号缓冲队列
2
i 号队列 "非"
3
是
否
4
轮询等待次数 C_i 清零
8
C_i=0
是
否
9
取出队头包的长度 L
输出 T 字节
5
10
计算轮询等待次数 $C_i=[L/64]_{取整}$
是否整数包
是
否
6
14
$C_i = C_i - 1$
否
13
C_i=1
是
16
轮询等待次数 C_i 清零
是
11
C_i=0
否
7
将当前包输出完毕
15
轮询等待一次
17
输出队头的包
轮询等待一次
12
18
$i=[i+1]_{mod\ T}$

图 2

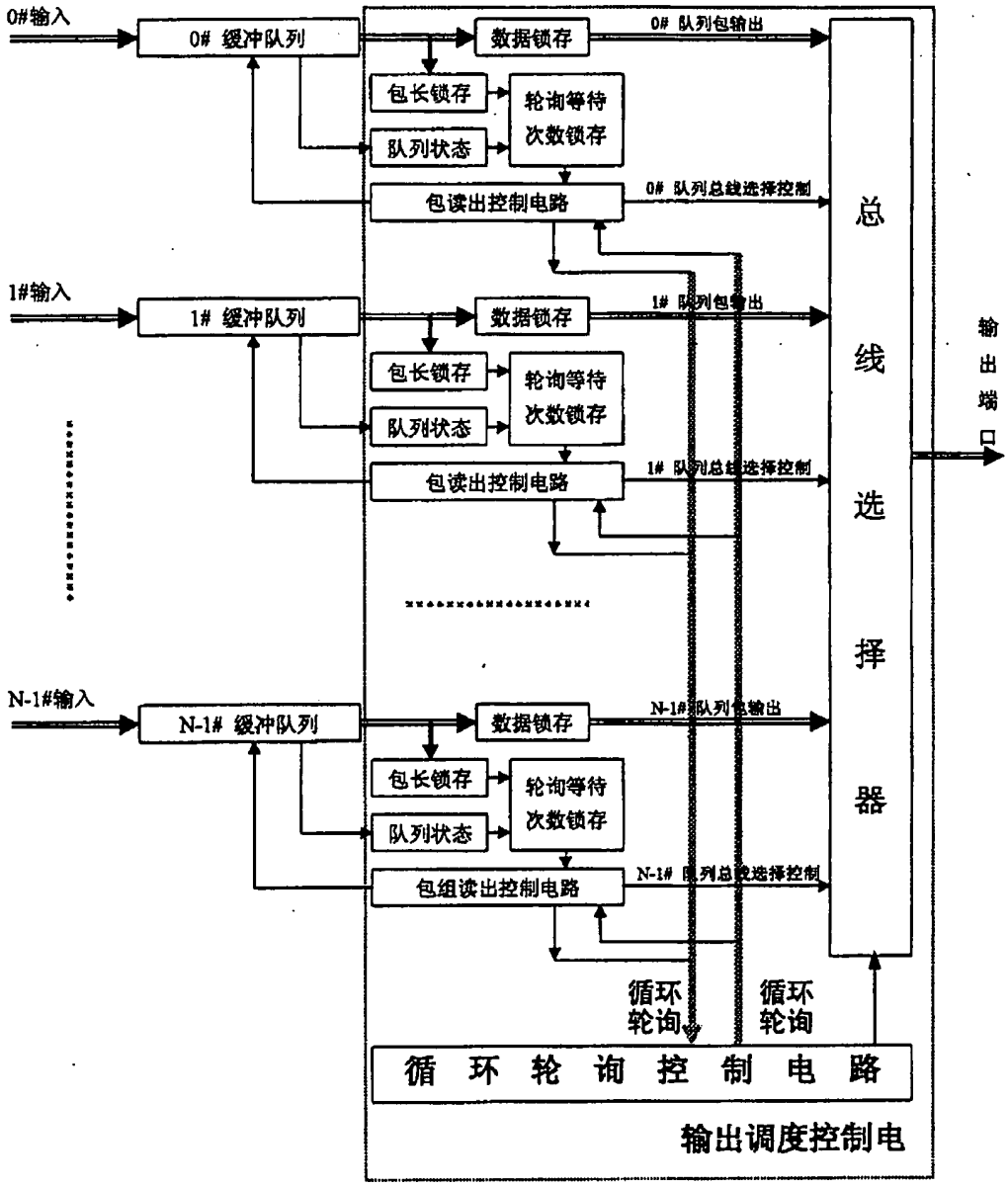

0#输入
0# 缓冲队列
数据锁存
0# 队列包输出
包长锁存
轮询等待
次数锁存
队列状态
包读出控制电路
0# 队列总线选择控制
1#输入
1# 缓冲队列
数据锁存
1# 队列包输出
包长锁存
轮询等待
次数锁存
队列状态
包读出控制电路
1# 队列总线选择控制
N-1#输入
N-1# 缓冲队列
数据锁存
N-1#队列包输出
包长锁存
轮询等待
次数锁存
队列状态
包组读出控制电路
N-1# 队列总线选择控制
总线选择器
输出端口
循环轮询
循环轮询
循环轮询控制电路
输出调度控制电

图 3

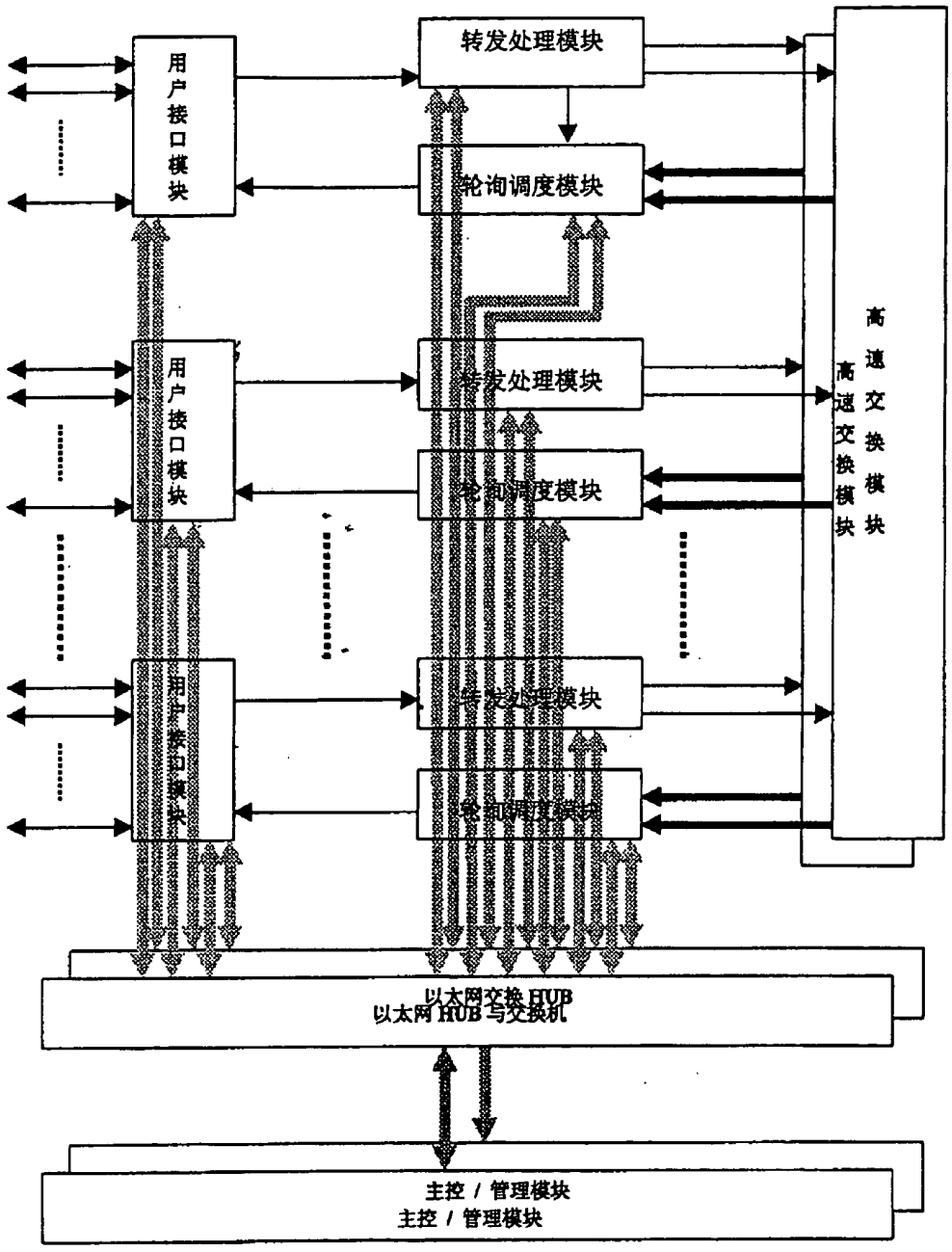


图 4

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☑ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.